Labels and Selectors

In Kubernetes, labels and selectors are powerful mechanisms used to organize, select, and manage Kubernetes objects like pods, services, deployments, etc.

**🏷️ Labels**

**🔹 What are Labels?**

**Labels** are key-value pairs attached to Kubernetes objects (like Pods, Services, ReplicaSets, etc.) that are used to **identify and organize resources**.

They are:

* Arbitrary: You can define any key-value pair.
* Non-unique: Multiple objects can have the same label.
* Immutable (mostly): Once applied, you usually don’t change them often (though they can be updated).

**🔹 Syntax**

Labels are defined as:

A black background with red and blue text

AI-generated content may be incorrect.

🔹 Example

A screen shot of a computer

AI-generated content may be incorrect.

**🎯 Label Selectors**

**🔹 What are Selectors?**

**Label selectors** are queries used to select a group of objects based on their labels.

They are used in many Kubernetes objects, such as:

* ReplicaSet selecting Pods
* Deployments selecting Pods
* Services selecting Pods
* …..

⚙️ Use Cases of Labels and Selectors

✅ Service targeting pods

A screen shot of a computer

AI-generated content may be incorrect.

➡️ This service will route traffic to all pods with label app: nginx.

✅ Deployment managing pods

A screen shot of a computer program

AI-generated content may be incorrect.

➡️ The Deployment uses the selector to manage only pods with the label app: nginx.

**🔍 Types of Label Selectors**

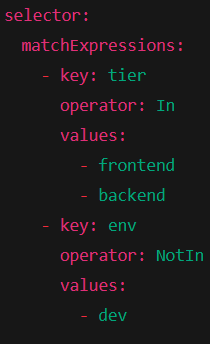
There are two types:

**1. Equality-based Selectors**

* Use =, ==, or !=
* Example:  
    
  A black screen with text

  AI-generated content may be incorrect.  
  Or in CLI:  
  

**2. Set-based Selectors**

* Use in, notin, or exists operators.
* **Example:**This selector matches resources where:
  + tier is either frontend or backend
  + env is **not** dev

